Post-placement Temperature Reduction Techniques

Wei Liu, Alberto Nannarelli
Technical University of Denmark
Kgs.Lyngby, Denmark

Andrea Calimera,Enrico Macii,Massimo Poncino
Politecnico di Torino
Torino, Italy

Abstract—**With technology scaled to deep submicron era, temperature and temperature gradient have emerged as important design criteria. We propose two post-placement techniques to reduce peak temperature by intelligently allocating whitespace in the hotspots. Both methods are fully compliant with commercial technologies, and can be easily integrated with state-of-the-art thermal-aware design flow. Experiments in a set of tests on circuits implemented in STM 65nm technologies show that our methods achieve better peak temperature reduction than directly increasing circuit's area.**

I. Introduction

Increased chip temperatures can have dramatic impacts on several figures of merit. Circuit behavior is strongly affected by temperature: MOS current drive capability decreases approximately 4% for every 10 °C temperature increase, and interconnect delay increases approximately 5% for every 10 °C increase. Thus, temperature variations across the die can result in significant timing uncertainties, requiring larger timing margins and lowering circuit performance. Elevated temperatures are also a major contributor to reduced reliability due to effects such as electro-migration or NBTI [1], [2]. Finally, the positive feedback between leakage power and temperature further exacerbates the thermal problem. Therefore, accurate on-chip temperature analysis and adequate thermal management are very important in deep submicron VLSI design.

High temperatures are caused by increased power density (i.e., power consumed per unit of area) caused by scaling. To reduce temperature (and thus power density) we need to either reduce total power consumption or increase area.

Although a vast literature on techniques to reduce power does exist (see [3], [4] for a survey), not all low-power design solutions are effective for temperature, given the large time constant of thermal events, which filters out the effects of most short-term power optimization solutions. For this reason, recent research has focused on specific solutions for *dynamic thermal management* ([5], [6]), in which temperature and not power is the actual metric.

Conversely, not many works have focused on smart management of area with the explicit objective of reducing power density. The only efforts have been at the micro-architectural, where some authors have explored the effects of floorplanning on the temperature distribution, and devising various thermal-aware floorplanning strategies [7], [8].

However, the same type of approach has not been investigated in the context of standard-cell designs in traditional synthesis flow. One possible reason is in a traditional back-end design flow, a potential increase in area means increasing chip cost and reducing yield. As a result, most floorplaning and placement tools try to place cells as compact as possible; this is also made possible by the fine grain of the atomic elements of placement, i.e., library cells. In modern design, the outline of a die is usually fixed while the component blocks and cells can be placed in a variable shape [9]. With the total cell area unchanged, this means we have some whitespace or area slack that can be exploited to alleviate the thermal problem.

Even a straightforward use of this area slack (e.g., by decreasing the row utilization factor during placement) would result in a decrease in cell (and, in turn, power) density over the entire circuit. Such a generalized, "blind" allocation, ignores the fact that peak temperature usually occurs in local hotspots which are groups of cells having larger switching activities than the rest of the circuit. Consequently, it would be desirable to reduce cell density mostly in the hotspots, while maintaining (or even increasing) cell density in cooler areas. In other words, we want to use the area overhead in regions that have higher temperatures.

In this work, we propose two approaches, *empty row insertion* and *hotspot wrapper*, for implementing a smart management of this additional area in such a way that peak temperature and temperature gradients can be reduced. Both methods can be easily integrated into mainstream placement tools. The two methods differ in the type of granularity at which the white space is allocated. In the former scheme, empty layout rows are inserted in proximity of hotspots, whereas in the latter individual cells are used to "wrap" the hotspot.

Compared with other thermal aware techniques, we work in a post-placement stage where we can exploit both functional information (i.e. the actual switching activity) and physical information (i.e. cell position) of the circuit so as to exactly localize the thermal hotspots.

Results show that a smart, hotspot-driven allocation of area can improve over a generalized one, especially for the case of small distributed hotspots.

II. Overview of Thermal Models and Analysis

We use the model proposed in [10], which consists of a conventional RC model of the heat conduction paths around each thermal element. The differential equation modeling heat transfer according to Fourier's law is solved by first transforming it into a a difference equation, and using SPICE to solve the equivalent RC electrical network. Since the model

978-3-9810801-6-2/DATE10 © 2010 EDAA

is not the focus of this paper, we only summarize in the following some relevant features of the thermal model.

Fig.1 shows the RC equivalent model and the geometrical structure for a thermal cell.

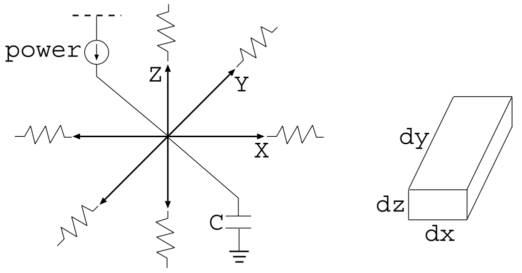

Fig. 1. Equivalent Model of a Thermal Cell.

The circuit is meshed into these thermal cells. Cells inside the circuit are connected to each other while cells on the boundary are connected to voltage sources which model the ambient temperature. Since the thermal time constant is in the order of tens of milliseconds, which is much larger than the clock periods in nanoseconds, we can neglect transient currents and solve the equation at the steady state; therefore, the capacitor in the cell model in Fig.1 can be removed and the SPICE netlist becomes a netlist of resistors, current sources and voltage sources.

The most important feature, however, is the definition of thermal cells in the context of a standard cell design. Since the positions of standard cells may not align exactly with thermal cells, we group several standard cells into one thermal cell. Thus, the power value in a thermal cell is the sum of power consumptions in all the standard cells that it covers.

Furthermore, temperature profile inside a chip is largely dependent on the package. For the same total power, it is possible to have different peak temperature and temperature gradient by using cooling mechanisms with different heat removal capabilities. In our thermal model, we adopted the thermal conductivities of different layers from [11]. The z direction is discretized into 9 layers and on each layer x and y directions are both discretized into 40 units which results in a grid of 1600 cells. For the size of the circuits we used in our experiments, this implies that a measuring point covers less than 10 standard cells. This provides us accurate temperature estimations at standard cell level.

III. Proposed methods

In this section we describe the two proposed schemes, *empty row insertion* and *hotspot wrapper* as post-placement temperature reduction techniques. Both methods aim to reduce the power density in the hotspot regions, by reducing cell density while keeping (cell) power consumption unchanged. They work on synthesized and placed design, and can therefore exploit detailed spatial information about the cells, besides using accurate, post-layout estimates of area, delay, and power. Figure 2 shows the flow of our methodology. On the left side of the flow, the synthesis and thermal/power estimation steps; the thermal simulator receives, as inputs, the placed netlist, (and therefore, the information about the cell positions and distances) the cell-by-cell power consumption information, and, not shown, the data relative to the process and the package. The thermal simulator builds the RC thermal network and solves using SPICE, and returns a thermal map of the die. The initial thermal map, together with the placed netlist info and a user-specified area overhead, are processed by our area management tool, which, using one of the two strategies, yields a modified placed netlist with better thermal properties.

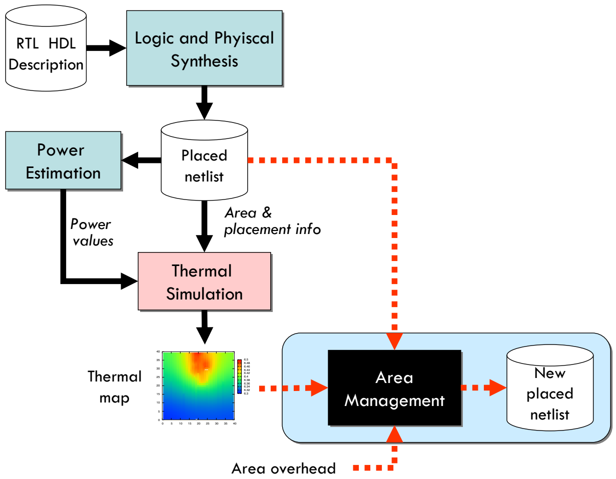

Fig. 2. Synthesis and Post-Layout Flow of the Proposed Methodology.

In both methods, the available area overhead is filled with *dummy cells* which do not contain active transistors and consume zero power. They can guarantee the electrical continuity of power and ground rails in each layout row. Dummy cells (also called filler cells) are also designed to meet all the design rules imposed by the technology (e.g. geometrical sizes and spaces, percentage of metal to guarantee a planar construction of stacked upper layers etc). This gives our methods a compliance with industrial semiconductor fabrication process. Moreover, the application of the proposed temperature reduction techniques does not limit the use of other thermal aware design methods. Instead they can be used as orthogonal methods which help to further reduce both peak temperature and temperature gradient.

A. Empty Row Insertion

Under this scheme, the granularity of the area slack insertion is a *layout row*. Conceptually it works as follows: In the area around a given hotspot, we insert an empty row between useful rows. This row of whitespace will be filled with *dummy cells*. In this way we increase the area only of the hotspot region. Since there is an empty row in every other row, the power density of the hotspot region is reduced evenly. Figure 3 shows such an example.

In the figure, rectangles denote standard cells while black areas denote whitespaces. The picture clearly shows that the bottom

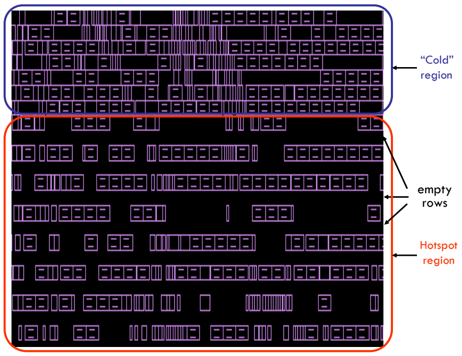

Fig. 3. Empty Row Insertion Example

part of the design contains empty rows alternated by "true" rows, while in the top part cells are placed in every row.

One advantage of this method is that it is easy to implement and integrate in back-end tools. Once we have identified the hotspots, we can easily move rows of cells upward by an offset of a few rows depending on how many empty rows have already been inserted.

Another advantage is that performance overhead is almost negligible if most local interconnections are between cells within the same row or among adjacent rows, which is often enforced in performance-oriented placement tools.

Furthermore a by-product of this transformation is that it increases the distance between rows of cells, thus reducing routing congestion in the hotspot regions.

The disadvantage of this scheme is that, due to the relatively coarse grain of the area increase, its efficiency depends on the *layout of the hotspot*. This is because when we insert an empty row, we increase the area of the entire row; If the hotspot is wide and involves most of the cells of the rows, then most of the introduced area overhead are used to reduce the power density of the hotspot. Otherwise if the hotspot is thin and tall, then most of the area is wasted.

B. Hotspot Wrapper

In this method, we insert filler cells one by one (i.e., not an entire row), that serve as a whitespace around a hotspot, which we call a *hotspot wrapper*. The placement tool tends to place cells in such a way that cell density is uniform across the entire chip. However, from the thermal point of view, it would be desirable that hotspot regions have lower cell densities.

Therefore, we isolate the hotspot from the rest of the circuit using a wrapper, namely, the cells which are the source of the hotspot are enclosed in a "whitespace ring". Once the hotspot is isolated, we reduce the cell density inside the wrapper by moving cells not belonging to the hotspot outside the wrapper and uniformly distribute the remaining cells in the wrapper area. In this way we reduce the power density only in well defined layout regions.

An example of the hotspot wrapper method is illustrated in Figure 4. The left figure shows that three hotspots are isolated and other cells have been moved outside to reduce the number of cells in the hotspot region. In commercial physical design tools, this can be done by creating exclusive move bounds which will force cells belonging to other units placed outside the specified region. The figure on the right shows the layout after we evenly redistribute the ”hot cells” so that they are not closely grouped together. Since changes of cell positions are local, performance overhead is very small if not negligible.

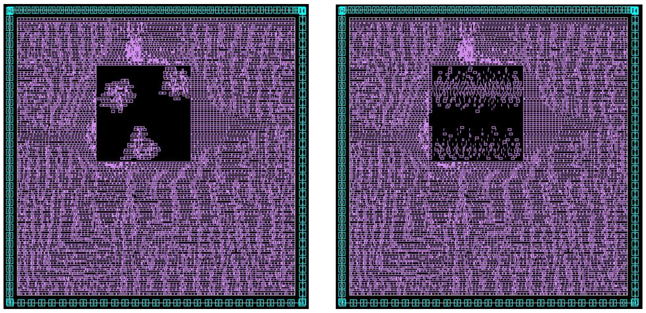

Fig. 4. Hotspot wrapper example

As can be seen this method is particularly useful for small concentrated hotspots. However, pushing cells away could increase the power density in the surrounding area and potentially making these areas new hotspots. Therefore, careful analysis of the power density map is needed before applying this method.

IV. Experiment results and analysis

The experiments are done on a synthetic benchmark circuit which is synthesized using a Synopsys flow and that consist of about 12000 standard cells. The reason behind using a synthetic benchmark is that in this way we are able control the size and position of hotspots using different workloads. Specifically, the circuit is composed of nine arithmetic units of various sizes. Clock frequency is set to 1 GHz.

Peak temperature for different configurations in our experiments ranges from a few degrees to 25 degrees above ambient temperatures. Since we are more interested in the relative amount of temperature reductions, we didn't include absolute temperature values in the test results. As an example, the power (left) and thermal (right) profile of test set one is shown in Figure 5. As can be seen in the power profile, there is significant correlation between highly power consuming area and thermal hotspots

The tools used for our methodology are Synopsys' VCS for logic simulation, Design Compiler for logical synthesis, IC Compiler for floorplanning and placement and Power Compiler for power estimation based on annotated switching activity of randomly generated test vectors.

Our first set of experiments are based on the configuration as shown in the left part of Figure 5 that has four scattered small hotspots. Figure 6 summarizes the results.

The plot shows the temperature reduction (meant as reduction of the peak temperature in the circuit) versus the area overhead used. The *Default* curve refers to the case in which the area overhead is uniformly distributed over all the circuit, as

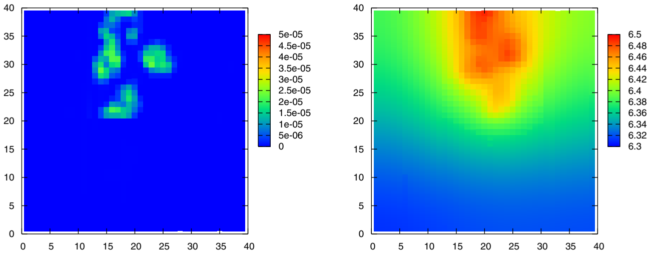

Fig. 5. Power and thermal profiles of test circuit

it happens when the utilization factor[1] during placement is reduced.

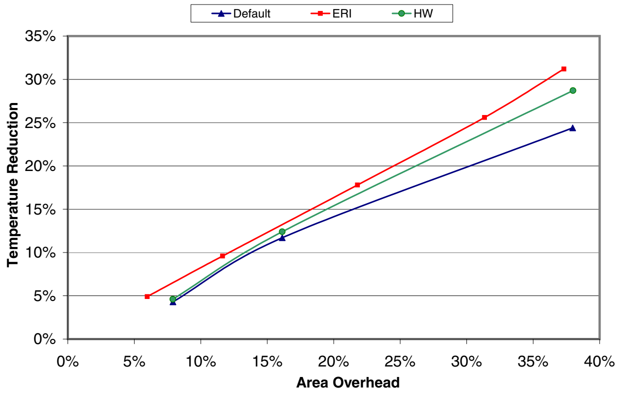

Fig. 6. Thermal Efficiency for the Various Techniques.

The *ERI* curve refers to the empty row insertion method. Notice that the points on the curve are not aligned with those of the "Default" curve because the area overhead for the two methods depends on different parameters. In the default case, it is achieved by relaxing the utilization factor; in the ERI one, it corresponds to different number of extra rows.

The *HW* curve refers instead to the hotspot wrapper case, for which the reference points on the curve are the same as in the default case. Under this scheme, in fact, we start from the default solution corresponding to a desired utilization factor, and perform the wrapper insertion on it.

From the plot we observe that both *ERI* and *HW* curves always lie above the default one, implying that both achieve higher temperature reductions for a given area overhead. In this specific example we can observe that *ERI* is better than *HW*, even if by a small amount. Finally, we can also notice that the effectiveness increases as the area overhead increases. The maximum timing overhead caused by applying the proposed methods is around 2%.

The second set of experiments refers to the case of a single, large, concentrated hotspot. Results are shown in Table I. Since the hotspot wrapper method is not suitable for large hotspot, we only compare the default scheme against the empty row insertion method. By inserting 20 rows in the hotspot region, we improve the temperature reduction by 13.1% (vs. 11.3%) using the same area overhead. Using 40 extra row, the benefit is even higher (28.6% vs. 20.2%).

	Area $[\mu m^2]$	*Inserted Rows*	*Area Overhead*	*Temp Reduction*
Default	361 × 361	–	16.1%	11.3%
Default	384 × 384	–	32.2%	20.2%
ERI	335 × 389	20	16.1%	13.1%
ERI	335 × 443	40	32.2%	28.6%

TABLE I
EXPERIMENT RESULTS BASED ON CONCENTRATED HOTSPOTS

V. CONCLUSION

Using the *empty row insertion* and *hotspot wrapper* methods we try to better utilize the extra whitespace that typically exist in modern designs to alleviate the thermal problem. Experiment results show that increasing the area in hotspots can effectively reduce the peak temperature. As future works, we would first like to find realistic benchmark circuits in order to test the proposed methods using real workloads. Another area of future research is to improve the efficiency of the approaches by transforming them into suitable optimization problems (e.g., the amount of empty rows or filler cells to be inserted).

REFERENCES

[1] Y.-K. Cheng, P. Raha, C.-C. Teng, E. Rosenbaum, and S.-M. Kang, "Thermal and power integrity based power/ground networks optimization," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 17, no. 8, pp. 668–681, Aug 1998.

[2] N. Kimizuka, et.al. "Impact of bias temperature instability for direct tunneling ultra-thin gate oxide on MOSFET scaling," *Symposium on VLSI Technology*, 1999, pp. 73–74.

[3] E. Macii, M. Pedram, F. Somenzi "High-level power modeling, estimation, and optimization," *DAC'97: 34th ACM/IEEE Design Automation Conference*, Jun 1997, pp. 504–511.

[4] J. Rabaey, "Low Power Design Essentials," *Springer*, 2009.

[5] D. Brooks and M. Martonosi, "Dynamic thermal management for high-performance microprocessors," in *High-Performance Computer Architecture, 2001. HPCA. The Seventh International Symposium on*, 2001, pp. 171–182.

[6] P. Liu, Z. Qi, H. Li, L. Jin, W. Wu, S.-D. Tan, and J. Yang, "Fast thermal simulation for architecture level dynamic thermal management," in *Computer-Aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference on*, Nov. 2005, pp. 639–644.

[7] W.-L. Hung, C. Addo-Quaye, T. Theocharides, Y. Xie, N. Vijaykrishnan, M. J. Irwin, "Thermal-aware floorplanning using genetic algorithms", *ISQED'05: International Symposium on Quality Electronic Design, 2005*, pp. 634–639, March 2005.

[8] K. Sankaranarayanan, S. Velusamy, M. Stan, K. Skadron, "A Case for Thermal-Aware Floorplanning at the Microarchitectural Level," *Journal of Instruction-Level Parallelism*, Vol. 8, 2005, pp 1-16.

[9] S. Adya and I. Markov, "Fixed-outline floorplanning : Enabling hierarchical design," *IEEE Trans. on VLSI*, vol. 11, no. 6, pp. 1120–1135, December 2003. [Online]. Available: http://www.gigascale.org/pubs/499.html

[10] W. Liu, A. Calimera, A. Nannarelli, E. Macii, and M. Poncino, "On-chip thermal modeling based on SPICE simulation," in *PATMOS '09: International Workshop on Power and Timing Modeling, Optimization and Simulation*, 2009.

[11] T. Sato, J. Ichimiya, N. Ono, K. Hachiya, and M. Hashimoto, "On-chip thermal gradient analysis and temperature flattening for soc design," in *ASP-DAC '05: IEEE Asia and South Pacific Design Automation Conference*, pp. 1074–1077, January 2005.

[1] Utilization factor is defined as total cell area divided by core area.